AMENDMENTS TO THE CLAIMS:

This Listing of Claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1.   (Currently amended) A computer-readable medium having embodied thereon a computer program configured to manage message queues used for transferring messages from a first system executing a first software application of an enterprise information technology system to a second system executing a second software application of the enterprise information technology system, wherein each message queue is used only for one object type, the medium comprising one or more code segments configured to:

receive an indication of an object type associated with a message independently of the message and the software applications, the object type including a category of enterprise application data;

identify a message queue used for the object type; and

perform a registration-related action on the identified message queue in response to the indication, the registration-related action affecting processing by middleware of messages stored in the identified queue and messages destined to the identified queue.

2.   (Previously presented) The medium of claim 1 wherein the one or more code segments configured to perform a registration-related action comprise one or more code segments configured to cause de-registration of the identified message queue such that processing of messages from the identified message queue is ceased.

3.   (Previously presented) The medium of claim 1 wherein the one or more code segments configured to perform a registration-related action comprise one or more code segments configured to cause registration of the identified message queue such that processing of messages from the identified message queue is started.

4. (Previously presented) The medium of claim 1 wherein the one or more code segments are further configured to perform a registration-related action to enable solving a problem with transferring enterprise application data having the object type to the second application.

5. (Previously presented) The medium of claim 1 wherein identifying the message queue comprises identifying a message queue used for the object type based on a name of the object type being included as part of a name of the message queue.

6. (Previously presented) The medium of claim 1 wherein identifying the message queue comprises identifying a message queue used for the object type by accessing a data structure having data that associates a name of the message queue and a name of an object type.

7. (Previously presented) The medium of claim 1 wherein the first software application comprises a sales system.

8. (Previously presented) The medium of claim 1 wherein a message includes enterprise application data.

9.   (Currently amended) A method for managing message queues used for transferring messages from a first system executing a first software application of an enterprise information technology system to a second system executing a second software application of the enterprise information technology system, wherein each message queue is used only for one object type, the method comprising:

receiving an indication of an object type associated with a message independently of the message and the software applications, the object type including a category of enterprise application data;

identifying a message queue used for the object type; and

performing a registration-related action on the identified message queue in response to the indication, the registration-related action affecting processing by middleware of messages stored in the identified queue and messages destined to the identified queue.

10.   (Original) The method of claim 9 wherein performing a registration-related action comprises causing de-registration of the identified message queue such that processing of messages from the identified message queue is ceased.

11.   (Original) The method of claim 9 wherein performing a registration-related action comprises causing registration of the identified message queue such that processing of messages from the identified message queue is started.

12.   (Currently amended) A system for managing message queues used for transferring messages from a first computer system, having a processor connected to a storage device and one or more input/output devices and executing a first software application of an enterprise information technology system, to a second computer system, having a processor connected to a storage device and one or more input/output devices and executing a second software application of the enterprise information technology system, wherein each message queue is used only for one object type and the processor of the second computer system is configured to:

receive an indication of an object type associated with a message independently of the message and the software applications, the object type including a category of enterprise application data;

identify a message queue used for the object type; and

perform a registration-related action on the identified message queue in response to the indication, the registration-related action affecting processing by middleware of messages stored in the identified queue and messages destined to the identified queue.

13.   (Original) The system of claim 12 wherein processor of the second computer system is configured to cause de-registration of the identified message queue such that processing of messages from the identified message queue is ceased.

14.   (Original) The system of claim 12 wherein processor of the second computer system is configured to cause registration of the identified message queue such that processing of messages from the identified message queue is started.

15. (Currently amended) A computer-readable medium having embodied thereon a computer program configured to manage message queues used for transferring messages from a first system executing a first software application of an enterprise information technology system to a second system executing a second software application of the enterprise information technology system, wherein each message queue is used only for one object type, the medium comprising a generic module with one or more code segments configured to:

receive an indication of an object type associated with a message independently of the message and the software applications, the object type including a category of enterprise application data;

receive an indication of registration-related action to be taken;

initiate, based on stored associations between object types and function modules, a specific one of the function modules for identifying a message queue used for the indicated object type and returning a queue name of the message queue used for the indicated object type;

when the indication of registration-related action to be taken is to register, register the message queue having the returned queue name such that middleware processes messages in the registered message queue; and

when the indication of registration-related action to be taken is to de-register, de-register the message queue having the returned queue name such that middleware ceases processing of messages in the de-registered message queue.

16. (Previously presented) The medium of claim 15 wherein the registration-related action enables solving a problem with transferring enterprise application data having the object type to the second application.

17. (Previously presented) The medium of claim 15 wherein a message includes enterprise application data.

Applicant : Ellen Kempin   Attorney's Docket No.: 13906-152001 / 2003P00627 US01
App. No. : 10/720,141
Filed   : November 25, 2003
Page   : 7 of 19

18.   (Currently amended) A method for managing message queues used for transferring messages from a first system executing a first software application of an enterprise information technology system to a second system executing a second software application of the enterprise information technology system, wherein each message queue is used only for one object type, the method comprising:

receiving an indication of an object type associated with a message independently of the message and the software applications, the object type including a category of enterprise application data;

receiving an indication of registration-related action to be taken;

initiating, based on stored associations between object types and function modules, a specific one of the function modules for identifying a message queue used for the indicated object type and returning a queue name of the message queue used for the indicated object type;

when the indication of registration-related action to be taken is to register, registering the message queue having the returned queue name such that middleware processes messages in the registered message queue; and

when the indication of registration-related action to be taken is to de-register, de-registering the message queue having the returned queue name such that middleware ceases processing of messages in the de-registered message queue.

19. (Currently amended) A system for managing message queues used for transferring messages from a first computer system, having a processor connected to a storage device and one or more input/output devices and executing a first software application of an enterprise information technology system, to a second computer system, having a processor connected to a storage device and one or more input/output devices and executing a second software application of the enterprise information technology system, wherein each message queue is used only for one object type and the processor of the second computer system is configured to:

receive an indication of an object type associated with a message independently of the message and the software applications, the object type including a category of enterprise application data;

receive an indication of registration-related action to be taken;

initiate, based on stored associations between object types and function modules, a specific one of the function modules for identifying a message queue used for the indicated object type and returning a queue name of the message queue used for the indicated object type;

when the indication of registration-related action to be taken is to register, register the message queue having the returned queue name such that middleware processes messages in the registered message queue; and

when the indication of registration-related action to be taken is to de-register, de-register the message queue having the returned queue name such that middleware ceases processing of messages in the de-registered message queue.

20. (Previously presented) The medium of claim 1, wherein the one or more code segments are further configured to:

receive an indication of an object type associated with a message independently of the message from a user.

21. (Previously presented) The medium of claim 1, wherein the middleware comprises software situated between the first software application and the second software application.

22. (Previously presented) The method of claim 18 wherein de-registering the message queue having the returned queue name comprises:

prohibiting messages destined to the de-registered message queue from being added to the de-registered message queue.

23. (Currently amended) A method for managing message queues used for transferring messages from a first system executing a first software application to a second system, other than the first system, executing a second software application, the method comprising:

receiving an indication of ~~an object~~ a document type associated with a message independently of the message and the software applications;

~~selecting, based on the object type, a function module for~~ identifying inbound and outbound message queues used for the ~~object~~ document type, the outbound message queue being located at the first system, the inbound message queue being located at the second system other than the first system, and the inbound message queue receiving messages from the outbound message queue; and

performing a registration-related action on each of the identified message queues in response to the indication, the registration-related actions affecting processing by middleware of messages stored in the identified message queues and messages destined to the identified message queues.

24. (Currently amended) The method of claim 23, wherein the receiving comprises:

receiving an indication of an object type ~~associated with a message independently of the message~~ from a user.

25. (Previously presented) The method of claim 23, wherein the middleware comprises software situated between the first software application and the second software application.

26.   (New) The method of claim 23, wherein the identifying further comprises:

identifying a hub message queue used for the document type and located at a message hub system, the hub message system receiving messages from the outbound message queue located at the first system and routing the received messages to the inbound message queue located at the second system.

27.   (New) The medium of claim 1, wherein the one or more code segments configured to identify a message queue comprise one or more code segments to:

identify a message queue used for the object type and located in a message hub system, the message hub system receiving messages from the first system and routing the received messages to the second system.

28.   (New) The medium of claim 1, wherein the type of enterprise application data includes one of a customer type, an employee type, an organization type, a business partner type, and a sales order type.

29.   (New) The medium of claim 1, further comprising one or more code segments to:

notify a user of the identified message queue; and

perform the registration-related action on the identified message queue after receiving a user confirmation.